

Development of An Intelligent Searcher

José Luis Carrillo Medina, Hugo Banda Gamboa, Members, IEEE
Systems Engineering Faculty, Escuela Politécnica del Ejercito Sede Latacunga,
Quijano y Ordóñez y Márquez de Maenza S/N Latacunga, Ecuador,
email: carrillojl@ltga.espe.edu.ec
Department of Informatics and Computer Science, Escuela Politécnica Nacional,
Isabel La Católica S/N Quito, Ecuador,
email: hbanda@ieee.org

Abstract. This article presents a description of the design and development of an Intelligent Searcher (a System based on an Intelligent Agent). In the first part there is a critic analysis related to the process for the construction of the agent based system, including three different methodologies. In the second part, the implementation and general results are discussed. The analysis presented here covers the entire software development life cycle.

Keywords: Agent-based system; Metadata - based searcher; Software agent development

I. INTRODUCTION

A. Agent

It is a computational system (hardware and software) characterized by the following properties:

- **Autonomy:** the agents operate without a direct intervention of human or other actor, and they have certain degree of control on their actions and their internal state;
- **Social ability:** the agents perform their actions together with other agents (and possibly with human beings) via some type of communication language among agents;
- **Reactivity:** the agents perceive their environment, (It can be the physical world, a user via a graphic interface, a collection of other agents, the internet, or perhaps a combination of all these), and then swiftly respond to changes that happen on it;
- **Proactivity:** the agents don't simply act in response to their environment, they are capable to exhibit opportune behavior, directed by objectives, taking initiative when it is appropriate;
- **Mental notions:** an agent has believes, desires and intentions;
- **Rationality:** an agent carries out actions in order to achieve objectives, without conflicting with other actors in their environment;
- **Truthfulness:** an agent cannot communicate false information on purpose;
- **Adaptability or learning.** An agent may have the ability to adapt or learn from its own experiences.

B. Neural Networks

An artificial neuronal network (ANN) is an attempt for simulating through computational procedures the behavior of part of the human brain. Concretely, an ANN is able to

analyze and reproduce the learning mechanisms and the recognition processes of an event that characterizes the evolved animals, like a human being.

C. Network Clustering (Classification)

Clustering is a process for grouping physical or abstract objects in classes of similar objects. It is required to define a similarity measure and then to apply it to form the classes. A class is a collection of objects whose similarity among them is high, and the similarity with objects of other classes is low.

D. Internet

The growth of the World Wide Web and the dynamic nature of the available resources in Internet constitute a challenge for the organizers of the information and it has been necessary to create tools to provide access to millions of documents that exist in electronic format. By means of search tools and using automatic methods to identify the resources of Internet, it has been given solution to these requirements. In fact, there is many programs that navigate automatically through the Web, looking for connections, recovering documents, indexing and creating databases. These systems find a great quantity of documents, but their precision is very low because an HTML document lacks enough information to describe the resource appropriately. This produces a lot of "garbage" as a result in the search for information. Internet, it is said, that grows in an exponential way, therefore it will be more and more difficult to navigate and to locate excellent information.

The necessity to organize and control the enormous diversity of information available in Internet has taken great efforts. Among these efforts there are two that stand out: the search motors that contain information of resources in Internet; and, the metadata - based searcher.

The first one is a tool that allows to access by means of keywords to the resources of Internet indexed in their database. Previous to this indexing, a selection and an evaluation of each one of the resources is done, using criteria established by each project.

The metadata – based searcher uses a group of informative elements to describe and represent objects of information.

This work describes the development of an intelligent searcher's prototype for Internet, using the metadata – based approach and the paradigm of software agents.

II. DESCRIPTION OF THE PROTOTYPE

The application developed provides the basic functionality required for a general purpose information filter. It comprises an agent that reads Web pages and an intelligent agent that helps the user to filter information. The user can specify a profile of key words and to provide explicit ranges for each web page. Two methods exist to filter the pages: key words and clusters. The user can select an option.

Two agents were created, one to pick up information from Internet web pages and other to qualify the information. The **URL Reader Agent** takes the data from the specified address and discharges the information. Then, each web page is qualified according to a list of keywords, using a neural network based **Filter Agent**.

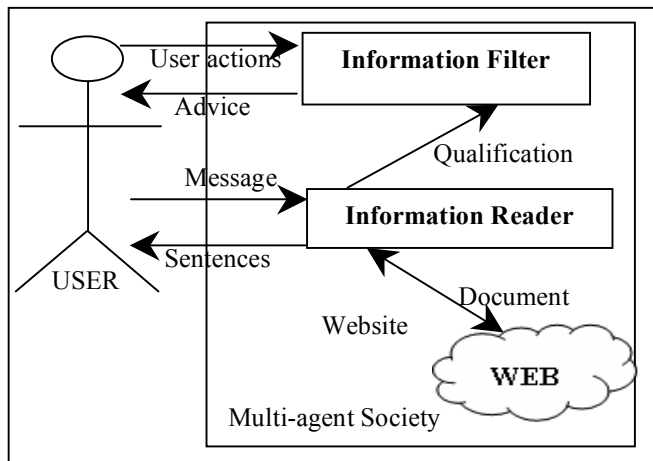


Fig. 1: Organizational Structures of the Multi-agent Society

A. Architecture of the Agent System

The architecture of the multi-agent system includes two types of different agents: an information filter of and an information Reader (Figure 1.)

B. Platform of Implementation

The object oriented programming tools were selected for the Win32 platform. This prototype was developed using the Java programming language.

C. Comparative Analysis of Methodologies

For the construction of the agent based prototype it was necessary to select a software development methodology. Two groups were considered: Object - oriented software methodologies and agent - oriented methodologies. For the selection of the methodologies it was taken into account the fulfillment of the following criteria:

- General purpose and open methodology
- Whole Life Cycle cover

- Strong interrelation within the development stages
- Availability of tools for the development
- Comprehensible notation
- Eclectic.

In the particular case of the agent - oriented methodologies they should take into account specific characteristics like:

- Intelligence of the agent
- Agent rolls and responsibilities
- Interactions among Agents.

According to fulfillment of these criteria, three methodologies were selected for the development of the agent – based prototype:

- Rational Unified Process, an Object - Oriented Methodology.
- MAS-CommonKADS, an Agent - Oriented Methodology.
- Agent Modeling Technique for BDI Agent Systems, an Agent - Oriented Methodology.

The Rational Unified Process was adopted because fulfilled the above mentioned indispensable conditions. According to the authors and our own experience in the modeling of object-oriented systems, this methodology is a result of the combination of the best characteristics and concepts of several other well known object - oriented methodologies (OMT, Booch, OOSE, etc.); and, it is also an open methodology.

MAS-CommonKADS [2] was selected as the agent - oriented methodology, since it belongs to the group of methodologies that are an extension of Knowledge Engineering methodologies. It is characterized because the first stages of the methodology possess a high level of abstraction allowing the identification of the initial components of the system in a general way. Additionally, it allows, in a deeper way, the specification of the intelligence characteristics of the agent, using methods and techniques of the Knowledge Engineering.

The Agent Modeling Technique for BDI Agent Systems was also applied. This one is regarded as the most complete in the group considered as an extension of the object – oriented paradigm for agent – based systems development. It comprises not only details related to mental states of an agent (desires, believes and intentions), agent rolls and different agent's responsibilities inside the application domain, but also the necessary interactions among the agents which are part of the system.

The prototype possesses all the typical characteristics included in the definition of an agent: autonomy, reactivity, sociability, pro-activity, mental notions, rationality and learning.

III. ANALYSIS OF PROCESSES AND RESULTS

After the development experience using the methodologies selected for this study, some significant points are worth mentioning:

A. The Unified Process Rational

In general, the methodology covers the whole life cycle of the software development. The gradual reduction of the level of abstraction during the process helps to the separation of specific concepts and the generality. The modularity is another strong point of this methodology, the transference from a conceptual model (architecture of modules) to a physical model (assignment of modules to processors) is simple and useful, since it ends with a conceptual view and the corresponding physical view of the system.

B. The MAS-CommonKADS Methodology

The methodology covers the aspects of the life cycle of an agent - based system. During the first stages, Conceptualization and Analysis, the methodology works at a high level of abstraction allowing the identification of components of the agent system. At the Design stage refines and deepens the abstraction. However, at the stages of Coding, Integration, Delivering and Maintenance it is rather complicated and indirect. This can be due, perhaps, to the fact that still does not exist a defined platform of agents for the implementation.

C. Agent Modeling Technique for BDI

This methodology does not cover all the aspects of the life cycle of an agent - based system. Its Analysis and the Design stages require a very high level of abstraction. Although particularly interesting for the deepest treatment of characteristics that it is possible to model, like mental notions, the resulting architecture is very abstract as be directly implemented in any development platform. For that to occur, it should be necessary to refine de design, adopting other methodology.

IV. TESTS OF THE PROTOTYPE

When executed, the web pages are loaded, ordered and grouped according to their qualification and level of interest for the user, by means of two filters:

- For the Keywords filter, the qualification is simply the sum of all the keywords (by default). The biggest the number of key words the more interesting it is.
- For the Cluster filter, the qualification is the average of the qualifications of the keywords of all the web pages that are inside the group. The higher the average of the group qualification the more interesting it is.

Before carrying out the classification it should be added the web pages of interest for training the neuronal network of the Cluster filter.

V. RESULTS

To test the prototype, in the first place, it was selected a group of keywords that will intervene in the classification of web pages:

Ord.	Keywords
1	Agente
2	Inteligente
3	Red
4	Neuronal

Then, four web pages HTTP addresses were entered. The system qualified them with the Keywords filter, for default. Next, it recovers and stores the web pages and executes the phase of training of the neural net. The new qualification is calculated applying the Cluster filter, obtaining the following results:

Keyword Filter

Ord.	Web Page Address	Qualification	Level
1	http://www.redcientifica.com/doc/doc19990331001.html	24	Interesting
2	http://www.acm.orgcrossroads/espaol/xrds5-4/dumbagents.html	15	Interesting
3	http://clib.telepolis.com/ohcop/neuretol.html	11	Interesting
4	http://www.sia.eui.upm.es/docent/ai.html	9	Normal

Table 1: Classification of four web pages (Filter: Keywords)

Interpretation

The web pages were classified by means of the Keywords Filter. In these cases the pages were ordered by the quantity of keywords found in its content, including its repetitions.

Cluster Filter

Ord.	Page Web	Qualification	Level
1	http://www.redcientifica.com/doc/doc199903310001.html	19.5	Interesting
2	http://www.acm.orgcrossroads/espaol/xrds5-4/dumbagents.html	19.5	Interesting
3	http://clib.telepolis.com/ohcop/neuretol.html	10.0	Interesting
4	http://www.sia.eui.upm.es/docent/ai.html	10.0	Interesting

Table 2: Classification of four web pages (Filter: Cluster)

Interpretation

The pages Web were classified by means of the Cluster Filter. In these cases the Web pages form two groups, which are conformed according to the quantity of similar words that have.

From the interpretations it is possible to see that the page number 1 has the biggest number of keywords, although these are repeated. Once trained the neural network in the

cluster method two groups are formed. The biggest qualification corresponds to the group formed by page number 1 and page number 2 (redcientifica and acm.org). Finally, from the combined results it can be deduced that the web page that the system recommends is redcientifica.

VI. DISCUSSION OF RESULTS

This application uses a simple method for exploration of data, where filter agents search for keywords. Only complete key words are counted. It could be possible to implement an algorithm to test for partial likeness so words in singular and plural were counted.

There also can be used other methods for classifying the keywords through neuronal nets.

VII. CONCLUSIONS

The object - oriented methodology can be used in the analysis, design and implementation of a software agent. In this work it has been demonstrated that it can be useful for the implementation of an agent oriented methodology.

A limitation found in the agent - oriented methodologies is the lack of CASE tools to support of the process of development of agent systems.

The agent based methodologies studied here turn out to exhibit many limitations:

- They only cover some aspects of the software development life cycle; or,
- They are centered in a particular architecture of a given agent.

As far as the application is concerned, the objective was to help users to realize how smartly an agent – based searcher can be applied to order hierarchically web pages based on its content.

VIII. REFERENCES

- [1] Booch, G., "Object Oriented Analysis and Design with Applications". The Benjamin/Cummings Publishing Company, 1994.
- [2] Iglesias, C. A., Garijo, M., González, J. C., and Velasco, J. R., "Analysis and design of multiagent systems using MAS-CommonKADS". In AAAI'97 Workshop on Agent Theories, Architectures and Languages, Providence, RI, ATAL, July 1997.
- [3] Jacobson, I., Christerson, M., Jonsson, P., and Övergaard, G., "Object-Oriented Software Engineering". A Use Case Driven Approach, ACM Press, 1997.
- [3] Kinny, D., Georgeff, M., and Rao, A., "A methodology and modelling technique for system of BDI agents". In W. Van der Velde and J. Perram, editors, Agents Breaking Away: Proceedings of the Seventh European Workshop on Modelling Autonomous Agents in a

Multy-Agent World MAAMAW'96. Springer-Verlag: Heidelberg, Germany, 1996.

- [4] Kruchten P., "The Rational Unified Process: An Introduction". Addison-Wesley Pub Co. 1998.

- [5] Wooldridge, M. and Jennings N. R., "Agent Theories, Architectures, and Languages: a Survey", in Wooldridge and Jennings Eds., Intelligence Agents, Berlin: Springer-Verlag, Vol. 1, N° 22, 1995.